



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

Am

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/038,892	10/22/2001	Terence Leong	SMQ-120/P6217	8871

959 7590 06/08/2005

LAHIVE & COCKFIELD, LLP.
28 STATE STREET
BOSTON, MA 02109

EXAMINER

HIRL, JOSEPH P

ART UNIT	PAPER NUMBER
----------	--------------

2129

DATE MAILED: 06/08/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/038,892	LEONG ET AL.	
	Examiner	Art Unit	
	Joseph P. Hirl	2129	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
 - If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
 - If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
 - Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 04 April 2005.
- 2a) ☒ This action is FINAL. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-60 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-60 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 04 April 2005 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____ |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

RD

DETAILED ACTION

1. This Office Action is in response to an AMENDMENT entered April 4, 2005 for the patent application 10/038,892 filed on October 22, 2001.
2. The First Office Action of September 27, 2004 is fully incorporated into this Final Office Action by reference.

Status of Claims

3. Claims 1, 12-14, 16-22, 30, 31, 33-37, 48-50 and 52-58 are amended. Claims 1-60 are amended.

Claim Rejections - 35 USC § 102

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

5. Claims 1-60 are rejected under 35 U.S.C. 102(e) as being anticipated by Fong et al (U.S. Patent 6,678,867, referred to as **Fong 67**; incorporated by reference to U.S. Patent 6,678,867 is U.S. 6,009,436 referred to as **Fong 36**; and incorporated by reference to U.S. Patent 6,678,867 is U.S. 6,085,196 referred to as **Fong 96**).

Claims 1, 12, 13, 25, 30, 37, 48, 49

Fong anticipates receiving a request to store in a database capable of storing data objects in a second programming language a first data object implemented in a first programming language including attributes and attribute values for a class (**Fong 96**, c 1, l 25-38); transforming the first data object to a second data object implemented in a second programming language, herein the second data object includes the attributes and attribute values of the class included in the first data object (**Fong 96**, c 1, l 25-38); and storing, in response to the request, the second data object to the database. (**Fong 96**, c 1, l 25-38; c 27, l 14-44; Examiner's Note (EN): para 13. applies; Fong does not establish a limit to the database size; Fong is user oriented to initiate the translation request and the transformed data object will be returned to the user via the GUI as a natural course of user involvement; computer implementation follows from Fig. 19).

Claims 2, 26, 38

Fong anticipates receiving a class schema including information on the class and attributes of the first data object (**Fong 96**, c 2, l 14-15; c 8, l 50-64); and using the received class schema to transform the first data object to the second data object (**Fong 96**, c 2, l 14-15; c 8, l 50-64).

Claims 3, 27, 39

Fong anticipates generating a source code file in the second programming language to implement the class and attributes included in the class schema (**Fong 96**, c 27, l 31, 32); compiling the source code file to generate an executable file that implements methods of the class (**Fong 96**, c 14, l 60); using one method of the class to construct the second data object (**Fong 96**, c 8, l 50-64); and including the attribute values from the first data object into the second data object (**Fong 96**, c 8, l 50-64).

Claims 4, 40

Fong anticipates generating statements into the source code file to define SET and GET interfaces for each attribute in the class (**Fong 96**, c 8, l 29-45; EN: SET establishes the attribute; GET actually "gets" the attribute value).

Claims 5, 41

Fong anticipates using at least one GET method in the first programming language to access the attribute values from the first data object (**Fong 96**, c 8, l 50-64); and using at least one SET method in the second programming language to set each attribute in the second data object to the corresponding accessed attribute value (**Fong 96**, c 8, l 50-64; EN: such occurs in the SGML to HTML mapping).

Claims 6, 42

Fong anticipates using at least one GET method in the first programming language to access the attribute values from the first data object (**Fong 96**, c 8, l 29-64); and generating statements into the source code file to set the attributes in the second

Art Unit: 2129

data object to the accessed attribute values from the first data object, wherein compiling the source code file produces the second data object with the attribute values set to the attribute values accessed from the first data object (**Fong 96**, c 8, l 29-64; c 27, l 31-44; EN: insertion into source code is merely software implementation and compiling typically follows to provide the binary instructions to drive the computer).

Claims 7, 28, 43

Fong anticipates the class schema includes for each attribute a name, data type and length of the attribute (**Fong 96**, c 8, l 13-64; EN: tags set beginning and end points).

Claims 8, 23, 44, 59

Fong anticipates the class schema is implemented in an Extensible Markup Language (XML) file (**Fong 67**, claim 4) .

Claims 9, 45

Fong anticipates wherein the database comprises an object oriented database (**Fong 96**, c 3, l 43-47).

Claims 10, 46

Fong anticipates wherein the first programming language comprises a non-Java object oriented language and wherein the second programming 3 language comprises the Java programming language (**Fong 96**, c 1, l 24-38; EN: JAVA is functionally similar to other object oriented programming languages as indicated in the specification on page 12, line 25).

Claims 11, 29, 47

Fong anticipates receiving a third data object implemented in the second programming language (**Fong 96**, c 4, l 25-26; EN: source to be mapped to itself or the second programming language); and adding the third data object to the database (**Fong 96**, c 27, l 14-43; EN: such is the database in memory).

Claims 14, 31, 50

Fong anticipates using a GET interface in the second programming language to access the attribute value in the accessed data object (**Fong 96**, c 8, l 29-45; EN: GET actually "gets" the attribute value); and using a SET interface in the first programming language to add each accessed attribute value from the accessed data object to the transformed data object (**Fong 96**, c 8, l 29-45; EN: SET establishes the attribute).

Claims 15, 32, 51

Fong anticipates receiving a request for at least one data object in the database from a second application that processes data objects in the second programming language (**Fong 96**, c 1, l 24-38; EN: such is the mapping process); accessing each requested data object from the database (**Fong 96**, c 1, l 24-38; EN: such is the mapping process); and returning each data object accessed from the database in response to the request from the second application to the second application (**Fong 96**, c 1, l 24-38; EN: such is the mapping process).

Claims 16, 33, 52

Fong anticipates providing a class schema, wherein the class schema includes information on one class and attributes of the class of the data object in the database,

wherein transforming the accessed data object to the transformed data object further comprises, for the accessed data object, using information on the attributes in the class schema for the class of the accessed data object to transform the accessed data object to the transformed data object (**Fong 96**, c 3, l 37-42; EN: such is the methodology of OOP).

Claims 17, 53

Fong anticipates accessing information on the length for each attribute in the class schema to generate the transformed data object with a size at least equal to a sum of lengths of all of the attributes in the class (**Fong 96**, c 3, l 43-47; EN: such would be a user establishing the linear units of storage space occupied by an object).

Claims 18, 34, 54

Fong anticipates determining whether the application requesting the data object processes data objects in the first programming language or the third programming language, wherein the step of transforming the accessed data object to the transformed data object implemented in the first programming language occurs if the application requesting the data object processes data objects in the first programming language (**Fong 96**, c 3, l 43-47; EN: such would be an object in SGML); transforming the accessed data object to the transformed data object implemented in the third language if the application requesting the data object processes data objects in the third programming language (**Fong 96**, c 3, l 43-47; EN: such would be an object in HTML); and returning the transformed data object in the third programming language to the

Art Unit: 2129

application that initiated the request (**Fong 96**, c 3, l 43-47; EN: such would be an object in HTML).

Claims 19, 35, 55

Fong anticipates receiving a definition of a class and attributes in the class (**Fong 96**, c8, l 14-64); generating a file (**Fong 96**, c 27, l 14-44; EN: files are typical means for data storage); and adding information on the class and each attribute in the received class definition to the generated file (**Fong 96**, c 27, l 14-44; EN: such is the process of information recording in a computer).

Claims 20, 36, 56

Fong anticipates adding information on a name, length and data type of each attribute in the class in the received class definition to the generated file (**Fong 96**, c 8, l 13-64; EN: tags set start and end of length).

Claims 21, 57

Fong anticipates generating at least one tagged element into the file including information on each attribute in the class (**Fong 96**, c 8, l 13-64).

Claims 22, 58

Fong anticipates (at least one) generating one tagged element into the file including information on a name of the attribute (**Fong 96**, c 8, l 13-64); generating one tagged element into the file including information on a length of the attribute (**Fong 96**, c 8, l 13-64; EN: length is defined by the start and end tag); and generating one tagged element: into the file including information on a data type of the attribute (**Fong 96**, c 8, l 13-64).

Claim 24, 60

Fong anticipates accessing the definition of the class, including information on attributes of the class, from a source code file of the class (**Fong 96**, c 8, l 13-64; c 14, l 60; EN: the source code are human readable program statements which contain class attributes).

Response to Arguments

6. The objections to claims 16, 33 and 52 under 37 CFR 1.75(c) are withdrawn.
7. The rejections to claims 14, 15, 17, 18, 30-32, 34, 50, 51, 53 and 54 under 35 USC 112, second paragraph, are withdrawn.
8. The rejections of claims 1-60 under 35 USC 101 as identified in paras 5. and 6. of the office action dated September 27, 2004 are withdrawn.
9. Applicant's arguments filed on April 4, 2005 related to Claims 1-60 have been fully considered but are not persuasive.

In reference to Applicant's argument:

Fong does not disclose receiving a request to store in a database capable of storing data objects in a second programming language a first data object implemented in a first programming language. The claimed invention receives a request to store a first data object implemented in a first programming language in a database that stores data objects implemented in a second programming language. The claimed invention transforms the first data object into a second data object that represents the class and class attributes of the first data object but in the second programming language, and stores the second data object in the database. In contrast, Fong describes a graphical user interface for creating user-defined mapping to convert a document in a first markup language to a document in a second markup language. As such, Fong is not concerned with storing a data object implemented in one programming language to a database that stores data objects in another programming language. Instead, Fong is focused on converting text-based documents from one text-based format to another text-based format. Furthermore, Fong does not discuss databases capable of storing data objects and storing data objects in such databases. Therefore, Fong does not disclose receiving a request to store a first data object implemented in a first programming language in a database that stores data objects implemented in a second programming language.

Examiner's response:

Para 13. applies. Fong 96 @ c 3, l 37-42 states:

It is a further object of this invention to provide a novel object-oriented system and computer program product for conversion of Standard Generalized Markup Language ("SGML") documents into HyperText Markup Language ("HTML") documents allowing a user to interactively define the mapping for the transformation.

Consequently, a plurality of programming languages exist, OOP is the framework, a computer system is involved, computers have memory and databases, mapping structure information in an object-oriented framework anticipates the applicant's invention since databases are involved in computer processing.

In reference to Applicant's argument:

Fong does not disclose receiving from an application that processes data objects in a first programming language a request for a data object in a database storing the data object in a second programming language. Rather, Fong describes creating a user-defined mapping to convert a text-based document in a first markup language to a text-based document in a second markup language. As such, Fong is not concerned with receiving a request from an application to obtain a data object from a database storing data objects in a programming language different than the programming language that the application processes the data objects. Instead, Fong describes converting text-based documents from one text-based format to another text-based format. Furthermore, Fong does not discuss databases capable of storing data objects and retrieving data objects from such databases. Therefore, Fong does not disclose receiving from an application that processes data objects in a first programming language a request for a data object in a database storing the data object in a second programming language.

Examiner's response:

Para 13. applies. The above response applies. Fong's disclosure is all about mapping structured information to different structured information in an object-oriented framework. Fong teaches the applicant's invention applied to a real world application.

In reference to Applicant's argument:

Art Unit: 2129

Fong does not disclose generating a file to provide a class schema representing the class and each attribute of the class of a first data object. As discussed above, Fong describes mapping strings of one character set of a document to strings of another character set. Fong does not describe generating a file and adding to the file class schema information. Instead, Fong describes changing the text-based format of an existing document. Furthermore, Fong is not concerned with providing a class schema to represent the class of a first data object. Rather, Fong is concerned with providing a document converted to a format specified by a user-defined mapping. Therefore, Fong does not disclose generating a file to provide a class schema representing the class and each attribute of the class of a first data object.

Examiner's response:

Para 13. applies. Applicant at claims 19, 35 and 55 is singularly concerned with one programming language, operating on a computer. In OOP, a generalized category that describes a group of more specific items called objects is referred to as a class. A schema defines aspects of the database, such as attributes (fields) and domains and parameters of the attributes. Fong in teaching the mapping of structured information to different structured information in an object-oriented framework will of necessity provide data objects, class of such objects and attributes since processing of such information in an object-oriented framework necessitates related database characterization. Applicant should review claims 19, 35 and 55 since these claims do not have the consideration of a plurality of languages and bear on a plurality of prior art that fundamentally underscores the concept of OOP.

Examination Considerations

10. The claims and only the claims form the metes and bounds of the invention. "Office personnel are to give the claims their broadest reasonable interpretation in light of the supporting disclosure. *In re Morris*, 127 F.3d 1048, 1054-55, 44USPQ2d 1023, 1027-28 (Fed. Cir. 1997). Limitations appearing in the specification but not recited in

the claim are not read into the claim. *In re Prater*, 415 F.2d, 1393, 1404-05, 162 USPQ 541, 550-551 (CCPA 1969)" (MPEP p 2100-8, c 2, I 45-48; p 2100-9, c 1, I 1-4). The Examiner has full latitude to interpret each claim in the broadest reasonable sense. Examiner will reference prior art using terminology familiar to one of ordinary skill in the art. Such an approach is broad in concept and can be either explicit or implicit in meaning.

11. Examiner's Notes are provided to assist the applicant to better understand the nature of the prior art, application of such prior art and, as appropriate, to further indicate other prior art that maybe applied in other office actions. Such comments are entirely consistent with the intent and spirit of compact prosecution. However, and unless otherwise stated, the Examiner's Notes are not prior art but a link to prior art that one of ordinary skill in the art would find inherently appropriate.

12. Unless otherwise annotated, Examiner's statements are to be interpreted in reference to that of one of ordinary skill in the art. Statements made in reference to the condition of the disclosure constitute, on the face of it, the basis and such would be obvious to one of ordinary skill in the art, establishing thereby an inherent prima facie statement.

13. Examiner's Opinion: paras 10.-12. apply. The Examiner has full latitude to interpret each claim in the broadest reasonable sense. Limitations appearing in the specification but not recited in the claim are not read into the claim. Applicant should review the full set of prior art identified in para 5. above together past art that defines OOP.

Conclusion

14. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the date of this final action.

15. Claims 1-60 are rejected.

Correspondence Information

16. Any inquiry concerning this information or related to the subject disclosure should be directed to the Examiner, Joseph P. Hirl, whose telephone number is (571) 272-3685. The Examiner can be reached on Monday – Thursday from 6:00 a.m. to 4:30 p.m.

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, Anthony Knight can be reached at (571) 272-3687.

Any response to this office action should be mailed to:

Commissioner of Patents and Trademarks,
Washington, D. C. 20231;

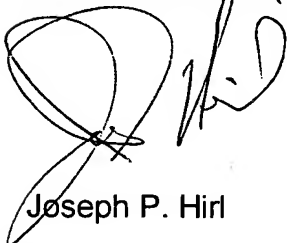
or faxed to:

(703) 872-9306 (for formal communications intended for entry);

or faxed to:

(571) 273-3685 (for informal or draft communications with notation of "Proposed" or "Draft" for the desk of the Examiner).

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have any questions on access to Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll free).



Joseph P. Hirl

May 31, 2005